

DYNAMIC METHODOLOGIES FOR PROBABILISTIC RISK ASSESSMENT (PRA)

Tunc Aldemir
The Ohio State University

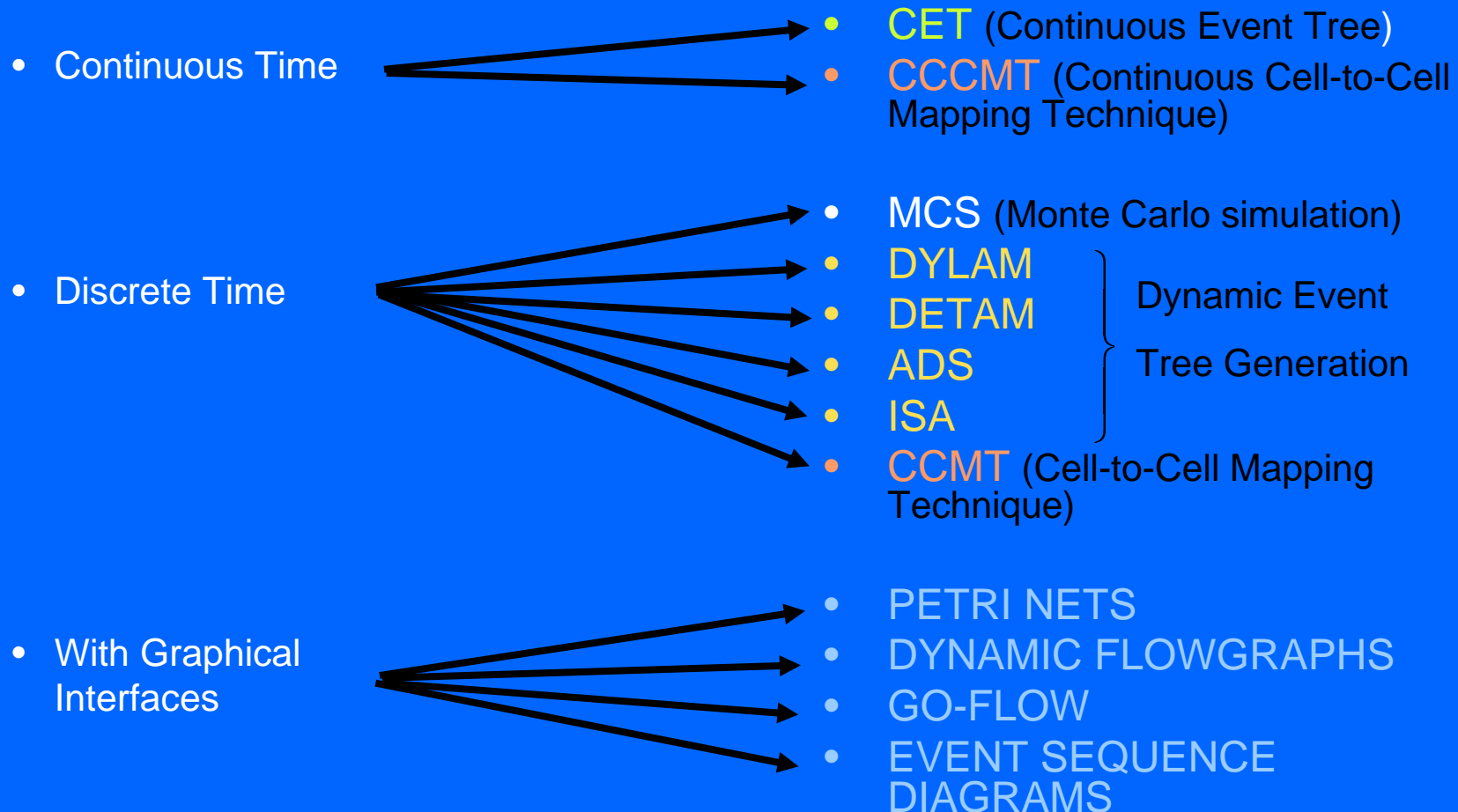
Background

- Conventional approach to probabilistic risk assessment uses the fault-tree/event-tree (FT/ET) methodology
- FT/ET at best can account for the order of occurrence of events in system evolution
- Dynamic methodologies are defined as those that explicitly account for the time element in probabilistic system evolution
- Dynamic methodologies are usually needed when the system has:
 - more than one failure mode,
 - control loops or indirect statistical dependence (coupling) of failure events through the controlled/monitored process (Type I coupling),
 - direct coupling of failure events through hardware/software (Type II coupling),
 - human interaction
- Dynamic methodologies are also expected to be needed for risk informed design of future reactors since uncertainties in model parameters may affect system behavior in a manner similar to component malfunction
- On-line application of dynamic methodologies necessitates identifying the current system state

Dynamic Methods for PRA

- Off-line applications
 - Prognostic methods
- Potential on-line applications
 - Diagnostic methods
 - Prognostic methods

Prognostic Dynamic Methods



Diagnostic Dynamic Methods

- Continuous Time \longrightarrow Adjoint CET
- Discrete Time \longrightarrow DSD

Prognostic Dynamic Methods - CET

- Describes the system behavior in terms of the probability $\pi(\mathbf{x}, i, t)$ of finding the system in the state-space (\mathbf{x} -space) with configuration i at a given time t .
- Input:
 - System trajectories $\mathbf{g}_i(t, \mathbf{x})$ in the state space
 - Configuration transition rates $\lambda_i(\mathbf{x})$ and $p(j \rightarrow i | \mathbf{x})$
 - Probability $F_i(t, \mathbf{x})$ that the system leaves configuration i before time t
 - Initial condition $\pi(\mathbf{x}, i, 0)$
- Output:
 - $\pi(\mathbf{x}, i, t)$
- Solution Method:
 - Monte Carlo in the integral form

$$\lambda_i(\bar{\mathbf{x}})\pi(\bar{\mathbf{x}}, i, t) = \lambda_i(\bar{\mathbf{x}}) \int \pi(\bar{\mathbf{u}}, i, 0) \delta(\bar{\mathbf{x}} - \bar{\mathbf{g}}_i(t, \bar{\mathbf{u}})) (1 - F_i(t, \bar{\mathbf{u}})) d\bar{\mathbf{u}} \\ + \sum_{j \neq i} \int_0^t \int \lambda_j(\bar{\mathbf{u}}) \pi(\bar{\mathbf{u}}, j, t - \tau) \left[\frac{p(j \rightarrow i | \bar{\mathbf{u}})}{\lambda_j(\bar{\mathbf{u}})} \right] \\ \times \delta(\bar{\mathbf{x}} - \bar{\mathbf{g}}_i(\tau, \bar{\mathbf{u}})) dF_i(\tau, \bar{\mathbf{u}}) d\bar{\mathbf{u}}$$

If Markov condition holds, i.e.

$$F_i(t, \bar{\mathbf{x}}) = 1 - \exp \left[- \int_0^t \lambda_i[\bar{\mathbf{g}}_i(s, \bar{\mathbf{x}})] ds \right]$$

where $\bar{\mathbf{g}}_i(t, \bar{\mathbf{x}}_0)$ is the solution of the i th dynamics

$$\frac{d\bar{\mathbf{x}}}{dt} = \bar{\mathbf{f}}_i(\bar{\mathbf{x}})$$

Then

$$\frac{\partial}{\partial t} \pi(\bar{\mathbf{x}}, i, t) + \text{div}(\bar{\mathbf{f}}_i(\bar{\mathbf{x}}) \pi(\bar{\mathbf{x}}, i, t)) + \lambda_i(\bar{\mathbf{x}}) \pi(\bar{\mathbf{x}}, i, t) \\ - \sum_{j \neq i} p(j \rightarrow i | \bar{\mathbf{x}}) \pi(\bar{\mathbf{x}}, j, t) = 0.$$

Prognostic Dynamic Methods - CCCMT

- Describes the system behavior in terms of the probability $\pi_i(j, t)$ of finding the system in the cell j of the state-space (\mathbf{x} -space) with configuration i at a given time t .
- Derivable from CET with

$$\pi_i(j, t) = \int_{j'} d\mathbf{x} \pi(\mathbf{x}, i, t)$$

- Input:**
 - Cell-to-cell transition probabilities $g(j|j', i', t)$ in the state space
 - Configuration transition rates $h(i|i', \mathbf{x}' \rightarrow \mathbf{x}, t)$
 - Initial condition $\pi_i(j, 0)$
- Output:**
 - $\pi_i(j, t)$
- Solution Method:**
 - ODE solvers

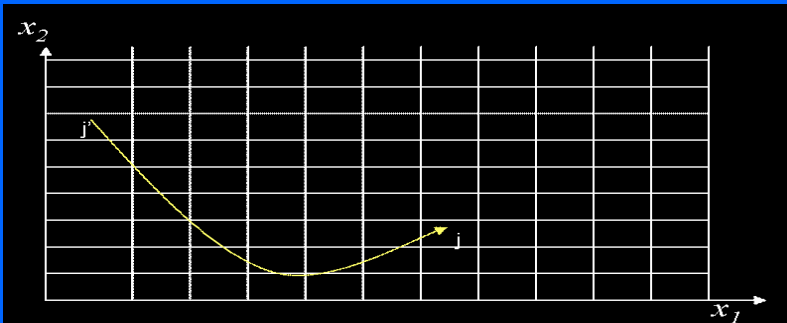
$$\frac{d\pi_i(j, t)}{dt} = \sum_{i'} \sum_{j'} [g(j|j', i', t) h(i|i', j' \rightarrow j, t) \pi_{i'}(j', t) - \lambda(i'|i, j', t) \delta_{jj'} \pi_i(j', t)]$$

$$h(i|i', j' \rightarrow j, t) = \frac{1}{v_{j'}} \frac{1}{v_j} \int d\mathbf{x}' \int_j d\mathbf{x} h(i|i', \mathbf{x}' \rightarrow \mathbf{x}, t)$$

$$\lambda(i|i', j', t) = \frac{1}{v_{j'}} \int d\mathbf{x}' \sum_{j'} \int_j d\mathbf{x} h(i|i', \mathbf{x}' \rightarrow \mathbf{x}, t)$$

$$g(j|j', i', t) = \begin{cases} -\frac{1}{v_{j'}} \int_{\substack{\hat{\mathbf{n}}(\mathbf{x}_s) \cdot \mathbf{f}(i', \mathbf{x}_s, t) > 0 \\ \mathbf{x}_s \in S_{j'}}} d\mathbf{x}_s \hat{\mathbf{n}}(\mathbf{x}_s) \cdot \mathbf{f}(i', \mathbf{x}_s, t) & \text{if } j' = j \\ \frac{1}{v_{j'}} \int_{\substack{\hat{\mathbf{n}}(\mathbf{x}_s) \cdot \mathbf{f}(i', \mathbf{x}_s, t) > 0 \\ \mathbf{x}_s \in S_{j'} \cap S_j}} d\mathbf{x}_s \hat{\mathbf{n}}(\mathbf{x}_s) \cdot \mathbf{f}(i', \mathbf{x}_s, t) & \text{otherwise} \end{cases}$$

$$\frac{d\mathbf{x}}{dt} = \mathbf{f}(i, \mathbf{x}, t)$$



Prognostic Dynamic Methods - CCMT

- Describes the system behavior in terms of the probability $\pi_i(j, k\tau)$ of finding the system in the cell j of the state-space (\mathbf{x} -space) with configuration i at a given time $k\tau$ ($k=0, 1, \dots$).

- Derivable from CET with

$$\pi_i(j, k\tau) = \int_{k\tau}^{(k+1)\tau} dt \int_{j'} d\mathbf{x} \pi(\mathbf{x}, i, t)$$

- Input:

- Cell-to-cell transition probabilities $g(j|j', l', t)$ in the state space
- Configuration transition rates $h(i|l', \mathbf{x}' \rightarrow \mathbf{x}, t)$
- Initial condition $\pi_i(j, 0)$

- Output:

- $\pi_i(j, k\tau)$

- Solution Method:

- Matrix solvers

$$\pi_i(j, k\tau) = \sum_{i'} \sum_{j'} g_{k-1}(j | j', i') h_{k-1}(i | i', j' \rightarrow j) \pi_{i'}[j', (k-1)\tau]$$

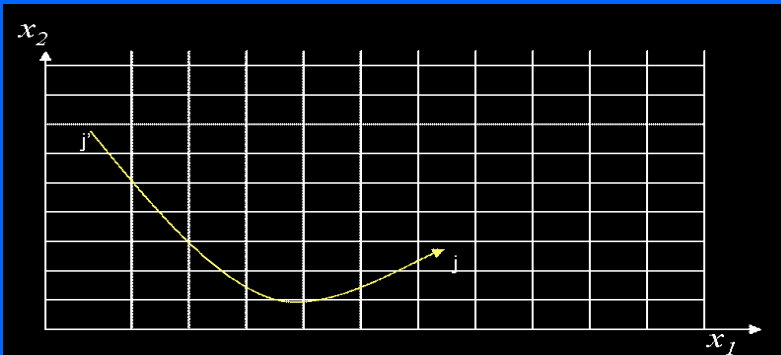
$$h_k(i | i', j' \rightarrow j) = \frac{1}{v_{j'}} \frac{1}{v_j} \frac{1}{\tau} \int_{k\tau}^{(k+1)\tau} dt \int_{j'} d\mathbf{x}' \int_j d\mathbf{x} h(i | i', \mathbf{x}' \rightarrow \mathbf{x}, t)$$

$$g_k(j | j', i') = \begin{cases} \frac{1}{v_j} \int_{j'} d\mathbf{x}' e_j[\tilde{\mathbf{x}}(i', \mathbf{x}', k\tau)] & \text{if } j' \text{ is within operating range} \\ 1 & \text{if } j \text{ is a failed state} \\ 0 & \text{otherwise} \end{cases}$$

$$e_j(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{x} \text{ is within } j \\ 0 & \text{otherwise} \end{cases}$$

$$\tilde{\mathbf{x}}(i, \mathbf{x}, k\tau) = \mathbf{x}[(k-1)\tau] + \int_{(k-1)\tau}^{k\tau} dt f[i, \mathbf{x}(t), t]$$

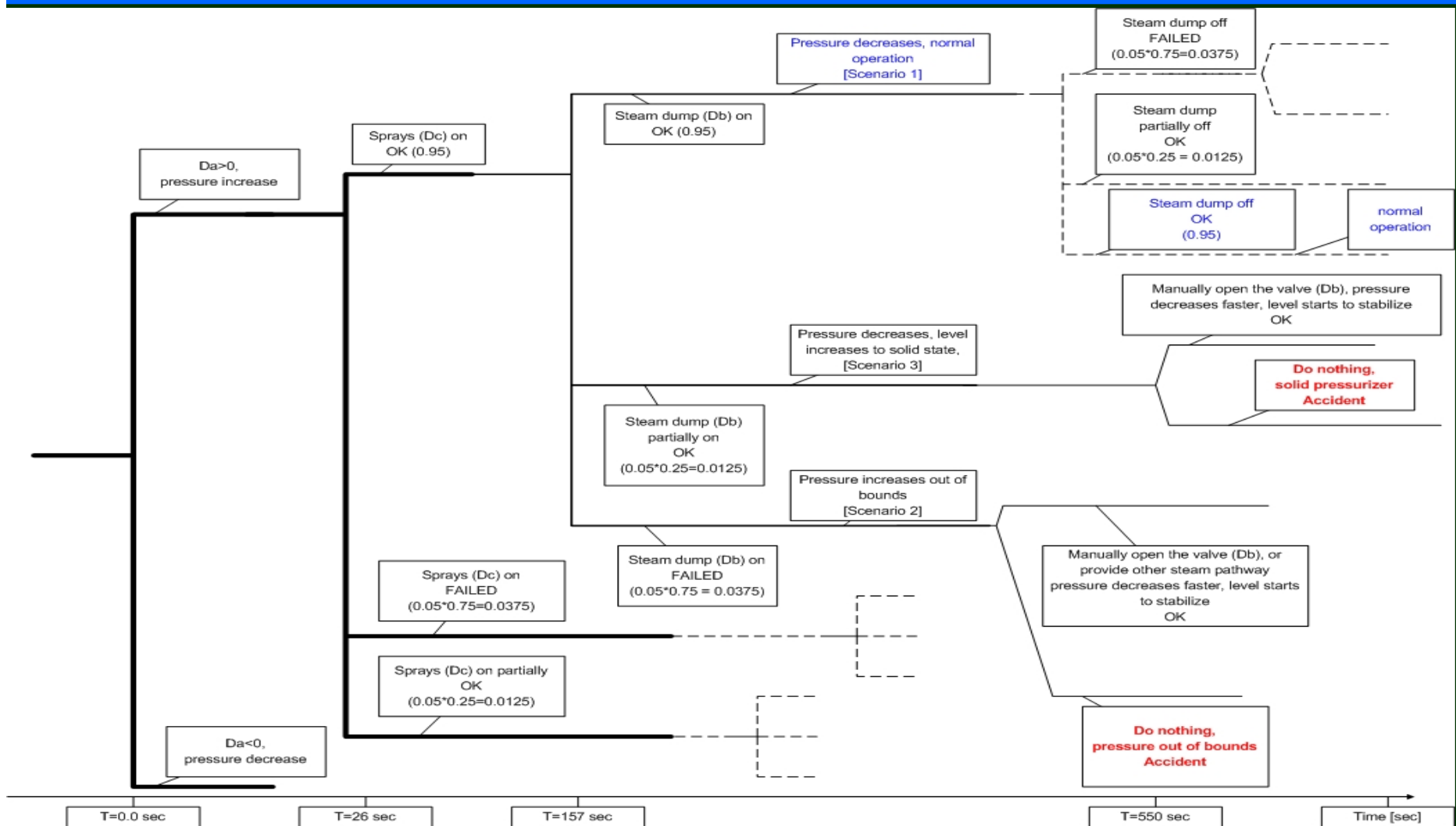
$$\frac{d\mathbf{x}}{dt} = \mathbf{f}(i, \mathbf{x}, t)$$



Prognostic Dynamic Methods – Dynamic Event Trees

- All methods generate event-trees based on possible trajectory branching during system evolution
- Methods differ in terms of branching and pruning rules, human reliability models and operator representation

Part of an ISA Event Tree for an Example Pressurizer



Prognostic DynamicMethods – Graphical

- Usually compatible with fault-trees
- Dynamic variables are represented as nodes of a graph
- Cause-effect relations are represented as the edges of the graph
- Models system evolution in terms of information transmission between nodes

Diagnostic Dynamic Methods – Adjoint CET

- Uses the “backward” Chapman-Kolmogorov equation to find the probability $\pi(\mathbf{x}_0, i_0, t_0 | \mathbf{x}, i, t)$ that the system was at location \mathbf{x}_0 and in configuration i_0 at time t_0 given that it is at location \mathbf{x} with configuration i at a given time t .
- Input:
 - System equations
 - Configuration transition rates $\lambda_i(\mathbf{x})$ and $p(j \rightarrow i | \mathbf{x})$
 - Initial condition $\pi(\mathbf{x}_0, i_0, 0 | \mathbf{x}, i, t)$ (or data from monitored variables)
- Output:
 - $\pi(\mathbf{x}_0, i_0, 0 | \mathbf{x}, i, t)$
- Solution Method:
 - Monte Carlo in the integral form

$$\frac{\partial \pi(\mathbf{x}_0, i_0, t_0 | \mathbf{x}, i, t)}{\partial t_0} + \mathbf{f}_i(\mathbf{x}_0, t_0) \cdot \nabla_0 \pi(\mathbf{x}_0, i_0, t_0 | \mathbf{x}, i, t) - \lambda_{i_0}(\mathbf{x}_0) \pi(\mathbf{x}_0, i_0, t_0 | \mathbf{x}, i, t) + \sum_{i \neq i_0} p(i_0 \rightarrow i | \mathbf{x}_0) \pi(\mathbf{x}_0, i_0, t_0 | \mathbf{x}, i, t) = 0$$

$$\frac{d\mathbf{x}}{dt} = \mathbf{f}_i(\mathbf{x}, t)$$

Diagnostic Dynamic Methods – DSD

- Uses a recursive Bayesian scheme adapted from the prognostic CCMT to determine the pdf $p(j_k|\hat{y}_k)$ of the system location j in the discretized state (or cell) and the configuration space at time $k\tau$ given the observation vector $\hat{y}_k = y_1 y_2 \dots y_k$ of observation y_k at each time point $k\tau$ (i.e. data from monitored variables).
- **Input:**
 - Observation y_k at each time point $k\tau$
 - Probability $p(x_k|\hat{y}_k)$ that the system is at location x within a cell at time $k\tau$, given the observation vector \hat{y}_k (quantifies uncertainty associated with the location of the system within the cell)
 - Probability $p(y_k|x_k)$ that the observation is y_k when the system is located at x at time $k\tau$ (quantifies measurement uncertainty)
 - Probability $p(x_{k+1}|x_k)$ that the system is at x_{k+1} at time $(k+1)\tau$, given that the system is located at x at time $k\tau$ (quantifies modeling uncertainty)
- **Output:**
 - $p(j_k|\hat{y}_k)$
- **Solution Method:**
 - ODE solvers to determine $g(j_{k+1}|j_k)$ if the system equations consist of differential equations

$$p(j_{k+1}|\hat{y}_{k+1}) = \frac{\sum_{j_k} g(j_{k+1}|j_k) p(j_k|\hat{y}_k)}{\sum_{j_{k+1}} \sum_{j_k} g(j_{k+1}|j_k) p(j_k|\hat{y}_k)}$$

$$g(j_{k+1}|j_k) = \iint_{j_{k+1} j_k} \frac{p(x_k|\hat{y}_k)}{\int_{j_k} p(x_k|\hat{y}_k) dx_k} p(y_{k+1} | x_{k+1}) \times p(x_{k+1} | x_k) dx_k dx_{k+1}$$

Current Projects at OSU Using Dynamic Methods

- Reliability modeling of digital instrumentation and control systems (NRC)
- Risk-based on-line accident management (SNL)
- Dynamic probabilistic extensions to SAPHIRE (INL)

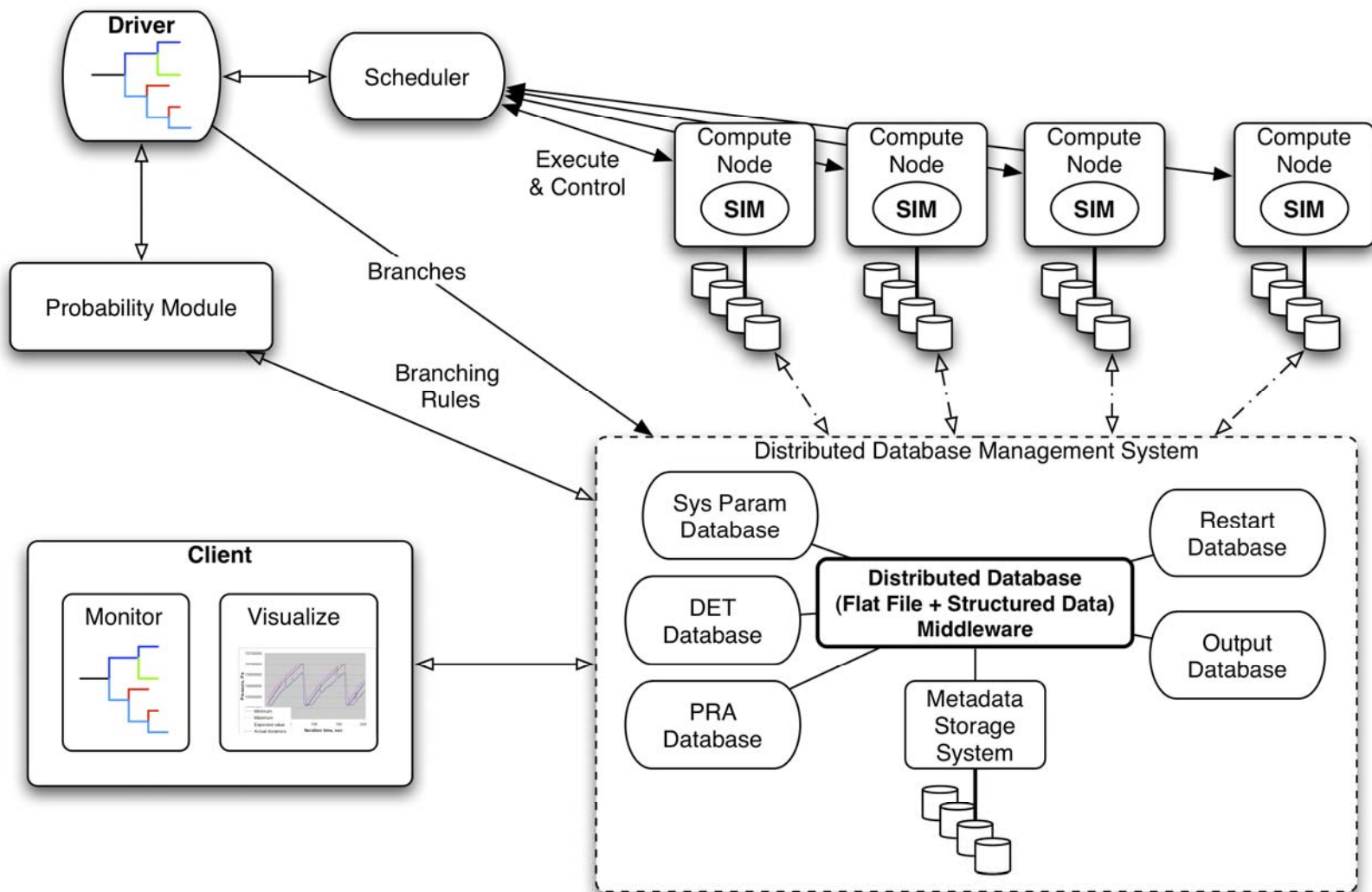
Reliability Modeling of Digital Instrumentation and Control Systems

- An objective is to develop a Markovian methodology for the reliability modeling of digital I&C systems
- Markovian methodology will use CCMT to describe the coupling between the digital I&C system failure events through the controlled/monitored process (Type I) as well as through direct communication and software (Type II)
- The resulting Markov transition matrix will be converted to dynamic event trees for incorporation into existing PRAs

Risk-based On-line Accident Management

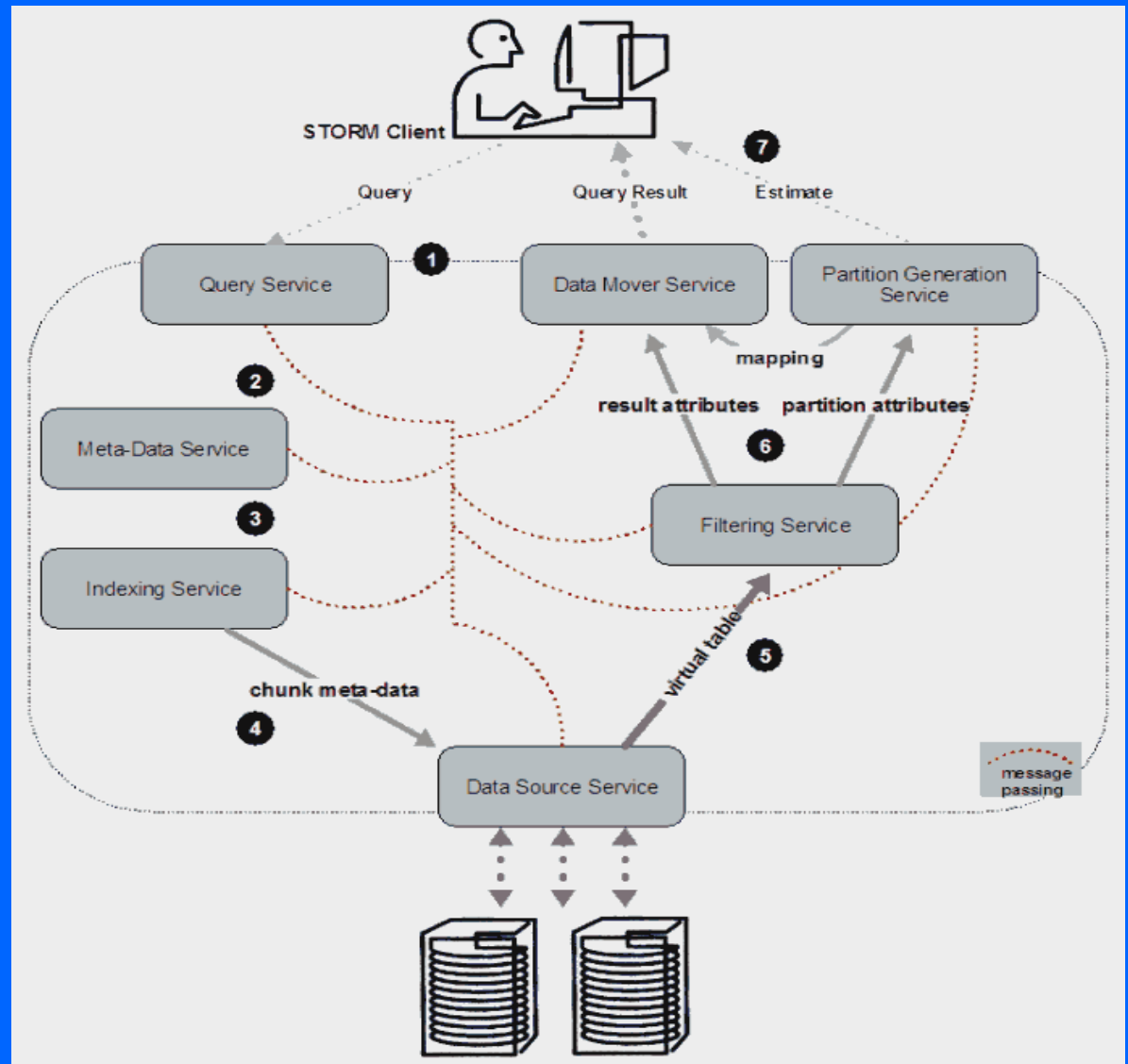
- An objective is to develop system independent software (driver) for mechanized generation of accident progression event trees (APETs) for Level 2 PRA
- Sample system analysis code being used is MELCORE
- Branching rules under consideration for passive components are based on fragility curves
- The driver is being designed for distributed computing
- Uncertainties in process modeling/data will be evaluated by coupling the driver to LHS software developed at SNL

System Architecture



System Architecture Summary

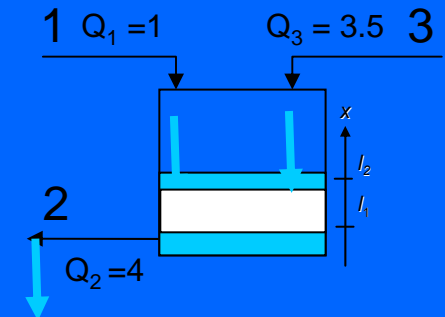
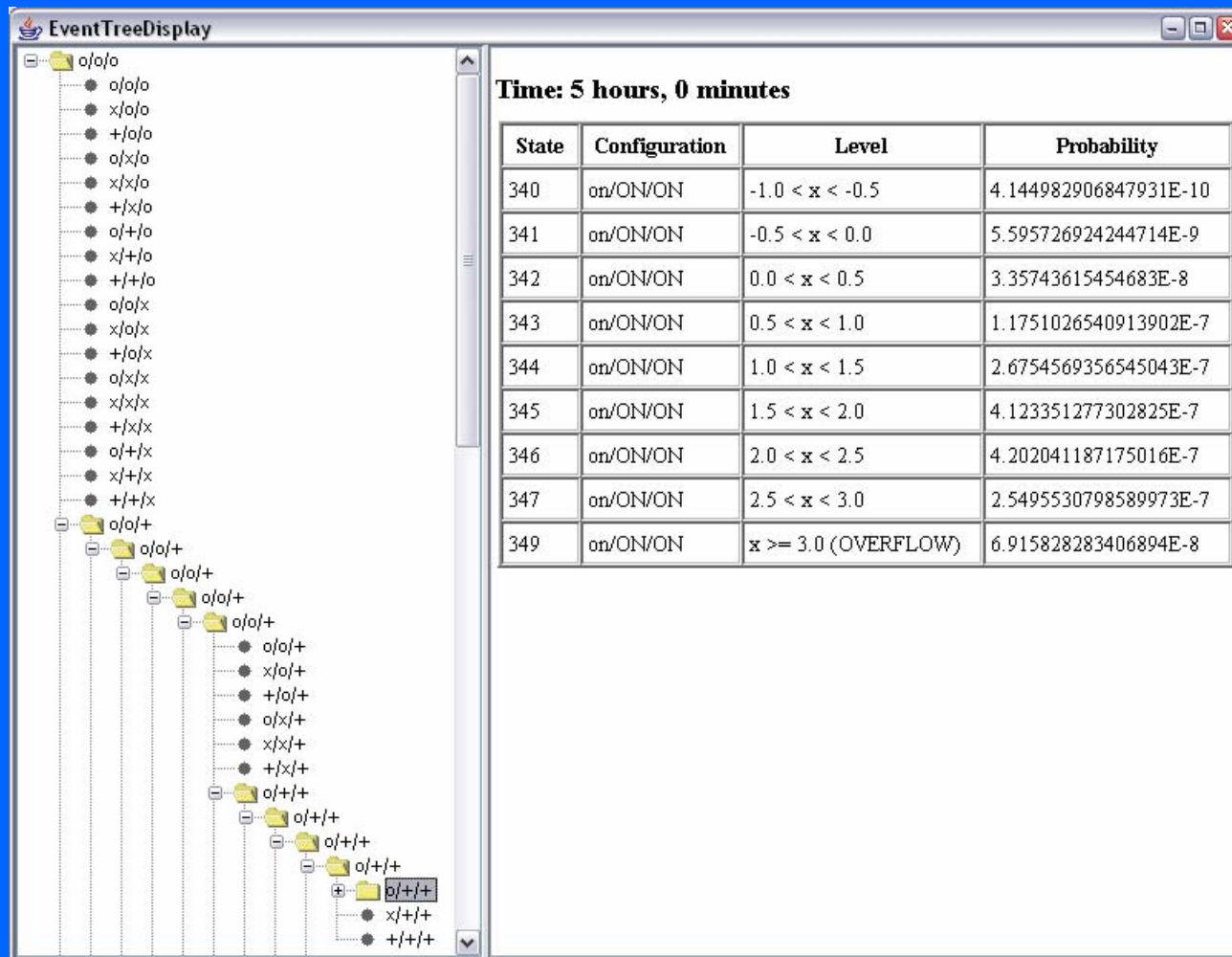
- Scheduler supports multiple execution backends
 - Condor, PBS, ssh/rsh
- Large output files retained on compute nodes
 - accessed by distributed STORM
- Only small files are stored in central database
 - System Parameters: Input files
 - PRA Database
 - DET Database: metadata about simulations



Dynamic Probabilistic Extensions to SAPHIRE

- SAPHIRE is a software tool to perform conventional fault-tree/event-tree analysis in a mechanized manner
- The project objective is to extend the applicability of SAPHIRE to systems where Type I and Type II coupling of failure events may be important
- An option to accomplish this objective is to develop modules on the SAPHIRE platform which can:
 - generate Markov models using CCMT, and,
 - convert the Markov model into dynamic event trees.

A Sample Dynamic Event Tree Generated from A Markov Model for a Simple Level Control System



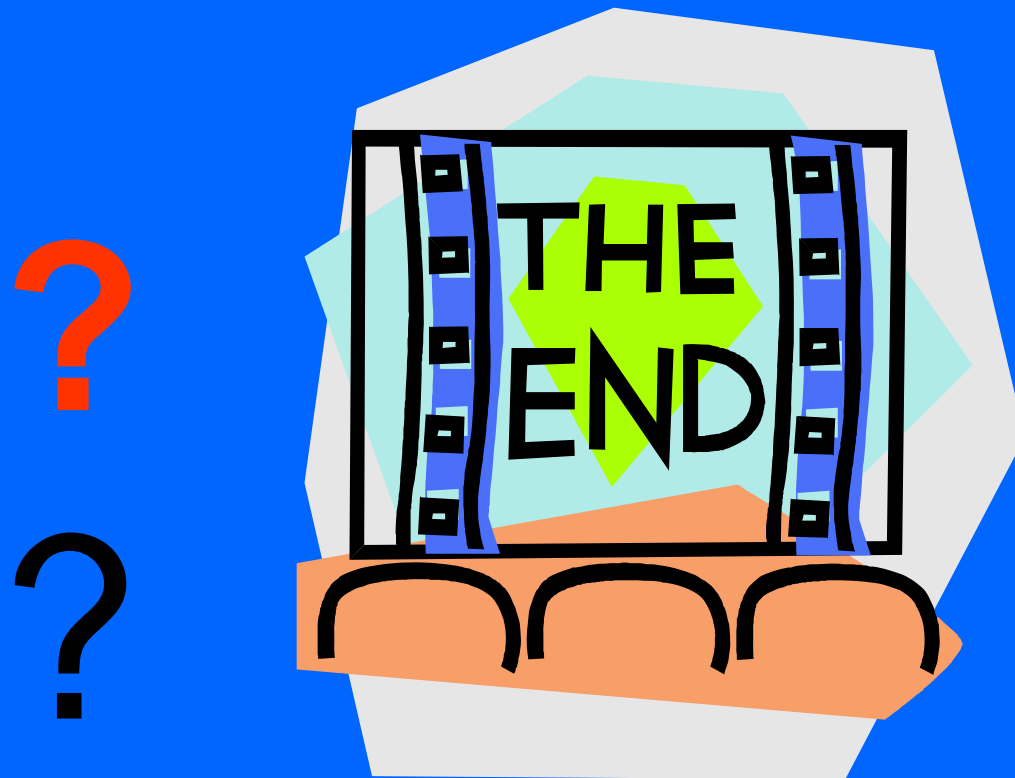
Normal operation

- If Unit 1 fails stuck when $x < l_1$, system fails by dryout
- If Unit 1 fails stuck when $x > l_2$, system fails by overflow

Conclusion

- Dynamic methodologies may be needed for PRA of systems with Type I and/or Type II coupling between failure events
- Dynamic methodologies are also useful for risk informed design/management of Generation IV reactors
- Dynamic methodologies may demand substantial computational resources
- Most dynamic methodologies are suitable for distributed computing

QUESTIONS ...



Thank you !!!